AD-A057 464 SCOTT POLAR RESEARCH INST CAMBRIDGE (ENGLAND) F/G 9/2

AN INTERACTIVE GRAPHICS SYSTEM FOR THE REDUCTION OF AIRBORNE LA--ETC(U)

1977 I J HOLYER, P WADHAMS, R T LOWRY N00014-76-C-0660

UNCLASSIFIED SPRI-TR-77-1 NL

| OF |

AD
A057 464

END
DATE
FILMED
9-78
DDC

12

SPRI

LEVEL II

SEA ICE GROUP

77-1

LASER ICE PROFILE ANALYSIS USING INTERACTIVE GRAPHICS

I.J.J. Holyer
Peter Wadhams
R.T. Lowry

1977

ADA057464

AD No.____
DDC FILE COPY

6

AN INTERACTIVE GRAPHICS SYSTEM FOR THE REDUCTION OF AIRBORNE LASER PROFILES OF SEA ICE.

11 1977

By

10

I. J. J. Holyer*

Peter Wadhams

12 29p.

Scott Polar Research Institute
Cambridge CB2 1ER, England

R. T. Lowry

Canada Centre for Remote Sensing,
2464 Sheffield Road, Ottawa, Canada

9 Technical rept.,

14

SCOTT POLAR RESEARCH ~~INSTITUTE~~ SPRI-TR-
TECHNICAL REPORT 77-1

15

Contract N00014-76-C-0660

New

*Present address:- Department of Pure Mathematics and Mathematical Statistics, University of Cambridge.

A

78 07 03 025

318 900

1B

CONTENTS

ABSTRACT

The authors have developed an interactive graphics system for the correction and reduction of airborne laser profiles of sea ice. The system is implemented on a Vector General renewable display system connected to a PDP11/45 computer, but minor modifications would make it suitable for any computer with interactive graphics. The laser profile is displayed in sections and the human operator is armed with routines for the removal of noise spikes, phase jumps and aircraft altitude variations. Completely automatic profile correction techniques have been found to be inadequate or unworkable, especially in heavily ridged ice. Although designed specifically to treat the output of the Geodolite laser profilometer, the system can be applied to many other data correction problems in geophysics and other sciences where long time series of data are obtained with periodic noise spikes or a long-period drift or modulation which requires removal.

B

1. INTRODUCTION

A knowledge of the surface topography of sea ice is of great importance in the modelling of polar ocean dynamics. The drag coefficient between wind and ice is determined by the surface roughness (Banke et al, 1976), and since most of the heat flow from ocean to atmosphere occurs through thin ice (Maykut, 1976) it is valuable to measure the distribution of surface elevations as a guide to the distribution of ice thicknesses. The airborne laser profilometer is the most effective instrument for swiftly generating such data over long stretches of the ice cover. It produces a profile of surface elevation along a track directly below the aircraft, using a beam whose footprint is essentially a moving point and which is therefore capable of very high resolution.

The earliest and most extensive laser surveys of the Arctic Ocean were carried out by the US Naval Oceanographic Office (Ketchum, 1971; Welsh and Tucker, 1971; Tooma and Tucker, 1973) and furnished data which were used to test theories of pressure ridge height and spacing distributions (Hibler et al, 1972, 1974; Diachok, 1975) and to delineate regions of the Arctic Ocean with differing degrees of roughness (Tucker and Westhall, 1973; Hibler, 1975). The same agency has carried out laser profiling in the Beaufort Sea as a contribution to the AIDJEX project, and co-operative profiling with a submarine in the East Greenland current (Kozo and Diachok, 1973). Laser profiling in the Arctic Ocean has also been carried out by an Argus aircraft of Department of National Defence, Ottawa (Dunbar and Lowry, 1974; Wadhams, 1976). As well as measuring ridging, it is also possible to use a laser profilometer to measure attenuation rates of ocean waves in the marginal ice zone (Wadhams, 1975). To date no laser profiling has been done in the Antarctic, although sea ice characteristics there differ significantly from those in the Arctic and a profiling experiment would be of great interest. The most recent laser profiling experiment in the Arctic Ocean has been a co-operative project in October 1976 involving an Argus aircraft and a British nuclear submarine following identical tracks at approximately the same time (Wadhams, 1977; Wadhams and Lowry, 1977), thus enabling the statistical relationship between surface and bottom topography to be derived. The interactive graphics system described in the present report was developed to analyse the data from this experiment.

So far all airborne laser surveys have employed the same instrument, the Spectra-Physics Geodolite 3A. This uses a continuous He-Ne laser beam with a wavelength of 632.8 nm and an output power of approximately 25 mW. The outgoing beam is sinusoidally modulated by an electro-optical modulator, and after reflection from the target the beam is received by a Schmidt-Cassegrain telescope of 0.2 m diameter which is coaxial with the laser. The phase shift between the modulations on the transmitted and received beams is measured, giving the range to the target. A set of modulation frequencies is available, usually differing by factors of 10. This enables the laser to be used as an accurate altimeter: the lowest modulation frequency gives a full scale deflection (0 to 2π in phase) for an altitude change of 10^5 feet, and the higher frequencies give 10^4, 10^3, 10^2 and 10 feet full scale deflection.

By switching frequencies stepwise, each significant figure of the altitude in feet can be obtained. When used to obtain sea ice terrain profiles only the higher frequencies are used, giving better resolution. Unfortunately, the aircraft normally porpoises with an amplitude exceeding 10 feet, and ice ridges can also exceed 10 feet in elevation, so that a laser profile on the 10 foot scale has frequent "phase jumps". A phase jump occurs when the range crosses one end of the 10 ft interval, ie the phase crosses 0° or 360°. When this happens the electronics automatically put in a 360° jump, positive or negative, in order to put the profile back on scale.

Figure 1 (top) shows a typical raw profile produced by this type of instrument. There are four components in the profile:-

1. The actual sea ice elevation.
2. A profile of aircraft porpoising, in which both altitude and attitude variations combine to give a smoothly-varying range oscillation of wavelength about 2-4 km. (ie Figure 1, centre).
3. Phase jumps, which are theoretically instantaneous and of identical amplitude. In practice, because of the recording system used (magnetic tape) and imperfections in the profilometer, the jumps have a finite rise time of up to 0.05 s and are of varying amplitude. The profilometer used for the 1976 experiment was modified to give a 20 ft full scale deflection on its most sensitive scale, reducing the frequency of phase jumps, and to give a jump of 180° instead of 360°. This restores the profile to the centre of the scale instead of to the opposite extremity, reducing the likelihood of the profile immediately jumping back by the same 360°.
4. Noise spikes and "drop-outs" (none visible on fig. 1). A drop-out is a brief positive-and-negative spike caused by a momentary loss of phase lock. The loss may be due to the aircraft crossing an ice-water boundary, or else experiencing an exceptionally sudden range change (eg an overhanging ice block in a ridge). Other spikes may be caused by interference from other equipment on the aircraft. Longer periods of noise can occur due to phase lock failure over thin cloud or undercast mist.

The purpose of a data processing system is first to remove (3) and (4), leaving a profile as in fig. 1 (centre), and then to remove aircraft porpoising as accurately as possible so as to leave a true profile of ice surface elevation as in fig. 1 (bottom). Computer programs can be written to automatically remove phase jumps, but the process is very unsatisfactory since the profiles must always be examined afterwards and sections with imperfect jump removal or with noise spikes present must be discarded or further treated. Similarly, there is no fully satisfactory way of automatically removing aircraft motion. A simple filter will not work, since the data do not represent a gaussian random process, the deviations from the porpoising being all in the positive direction. A three-stage filter was described by Hibler (1972) to overcome the last problem. The method used is to connect up the local minima, thus underlining the data. This underlining is then filtered and subtracted from the corrected data to yield an ice profile. The main disadvantages of this procedure are that it does not allow for the existence of leads, and again that it does not allow for interactive

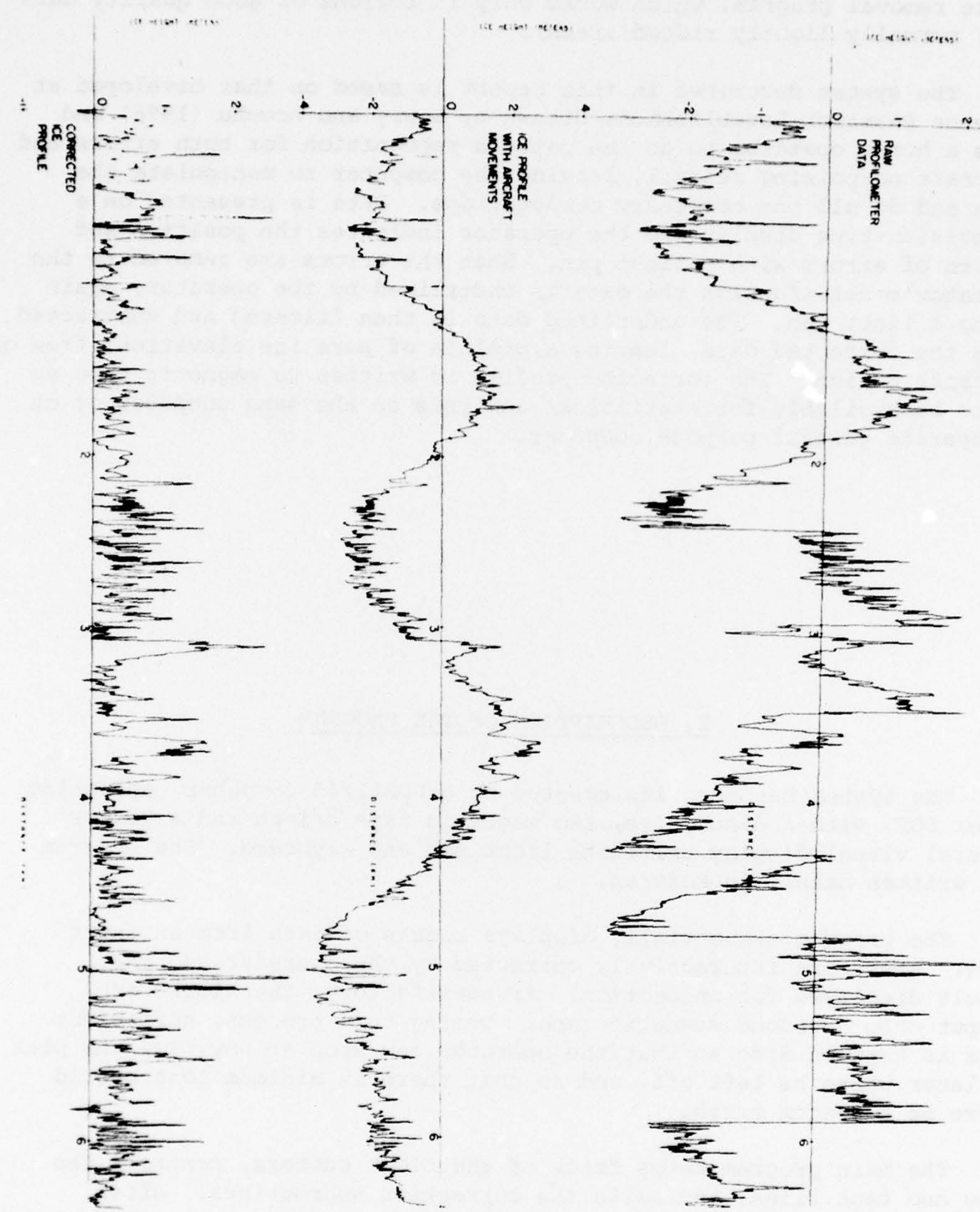

Figure 1. A laser profile of sea ice, (a) as recorded, (b) with phase jumps removed, (c) with aircraft porpoising removed.

viewing. Thus it must be coupled with an automatic phase jump/noise spike removal program, which works only in regions of good quality data (ie, normally lightly ridged areas).

The system described in this report is based on that developed at Defence Research Establishment Ottawa by Lowry and Brochu (1976) and uses a human operator to do the pattern recognition for both errors and aircraft porpoising removal, leaving the computer to manipulate the data and do all the necessary computations. Data is presented on a television-type display and the operator indicates the position and nature of errors with a light pen. When the errors are removed to the operator's satisfaction the data is underlined by the operator, again using a light pen. The underlined data is then filtered and subtracted from the corrected data, leaving a profile of pure ice elevation, free of aircraft motion. The corrected profile is written to magnetic tape so as to be available for statistical analysis on the same computer or on a separate general-purpose computer.

2. DESCRIPTION OF THE PROGRAM

The system has been implemented on a PDP11/45 computer, operating under DOS, with a disc drive, two magnetic tape drives and a Vector General visual display unit with light pen and keyboard. The program was written mainly in Fortran.

The program sequentially displays blocks of data from an input tape; these are interactively corrected by the operator and the result displayed for inspection. If satisfactory, the blocks are output onto a second magnetic tape. During this process, sufficient data is kept on disc so that the operator can stop at any time and pick up later where he left off, and so that there is minimum loss should there be a system crash.

The main program keeps track of the block numbers, controls the disc and tape files, and calls the correction subroutines. After asking the operator whether the data is new or old (ie system being restarted after a break) the program reads in the relevant block of data into the array ICE, which also holds subsequent corrected versions of the data. The program then hands this over to the subroutine CORREK for the correction of phase jumps and noise spikes, and then to the subroutine UNDERL for the drawing of the aircraft motion. This second routine produces an array PTS holding the drawn underline.

This underline, which consists of line segments, is filtered by a Hamming filter, which essentially produces a weighted running mean of 200 points. However, to avoid edge effects of the filter at the beginning and end of a block, the array PTS must be preceded by at least 100 points of the underline from the previous block (stored in array BSTR), and must be followed by the first 100 points of the subsequent block (stored in array ESTR). The very first and last blocks on the input tape must therefore have end ramps constructed for them. For this reason the filtering (subroutine HAMFIL) must take place a block behind the correction and underlining. Thus for new data the sequence of events is:-

-- process blocks 1 and 2
-- filter and display block 1
-- process block 3
-- filter and display block 2 etc.

When a block is filtered and displayed, the operator may reject it, in which case this block is treated as a first block of data and the program restarted, or he may accept it, in which case the smoothed underline is subtracted and the result similarly displayed for approval. When finally approved, this block is written onto output tape.

It should be noted that the program has been linked with three standard libraries of routines held permanently on the Cambridge PDP11/45 computer:-

1. A standard Fortran subroutine library
2. Magnetic tape routines

MTREAD	reads a block
MTWRT	writes a block
MTWTM	writes a tape mark
MTSKIP	skips blocks
MTRWND	rewinds

3. Graphics routines

NITDEV	initialise device
NITBUF	initialise a picture buffer
BGNPIC ENDPIC	for creating a picture segment
DELPIC	for deleting a picture segment
WRITOL	for positioning tracking square
INWAIT	for inputting on visual display unit/Vector General keyboard
REATOL	for finding position of tracking square
LINI LINE LINIR	absolute and relative line segment drawing
WINDW	for setting up a display window

The subroutine CORREK has two main functions. Firstly it seeks large jumps between consecutive points. It indicates these in turn, asking the operator to say :-

i) that it is a phase jump or noise spike, in which case it is removed, or
ii) that the jump is genuine data, in which case it is kept.

After this the routine waits for commands from the operator to remove remaining errors, redraw, reject and start again, or finally accept the modifications (see OPERATOR'S GUIDE).

The subroutine UNDERL waits for commands to insert or delete a line segment, alter the starting point of the line, or finally accept the underline.

A basic block diagram for the main program is shown in Fig 2. Block diagrams for the subroutines will be found in section 4.

Some modifications to the simplest form of the program were necessary because of space and speed limitations on the PDP11/45. The program fills most of the available 45K of store, so space saving procedures were employed which makes the program (section 4) slightly difficult to follow. Further HAMFIL as written in FORTRAN takes 90 secs for a filter, and so was replaced by an equivalent routine written in Assembler code, which requires only 15 secs.

The Hamming Filter was chosen for its low side lobes in the Fourier transform domain. It should be pointed out however, that a simple top hat filter could be used. Its first side lobe is only 13dB down, but in many cases this is entirely adequate. (If the data are to be subject to Fourier analysis, however, the side lobes become more important). A top hat filter, once filled, will filter any length of data block, with only a few simple calculations, thus taking much less time.

3. AN OPERATOR'S GUIDE TO THE SYSTEM

This section refers specifically to the operating system of the PDP11/45 - Vector General. It is included to illustrate the relative ease of carrying out these operations on an interactive graphics system. The program is set up on the user's disc; the data is in digital form on magnetic tape; the output is written to a second magnetic tape. Normally Geodolite data is recorded on analogue magnetic tape, so that A-to-D conversion is necessary. The 1976 data was recorded thus on a Tandberg Series 100 FM instrumentation tape recorder with flutter compensation, and digitally converted using an AD01-D analogue-to-digital converter built into the PDP11/45 computer. Digital data points were 0.01 s apart, corresponding to about 0.9 m of track; an analogue low-pass filter was included in the A-to-D converter to remove all energy above the Nyquist frequency.

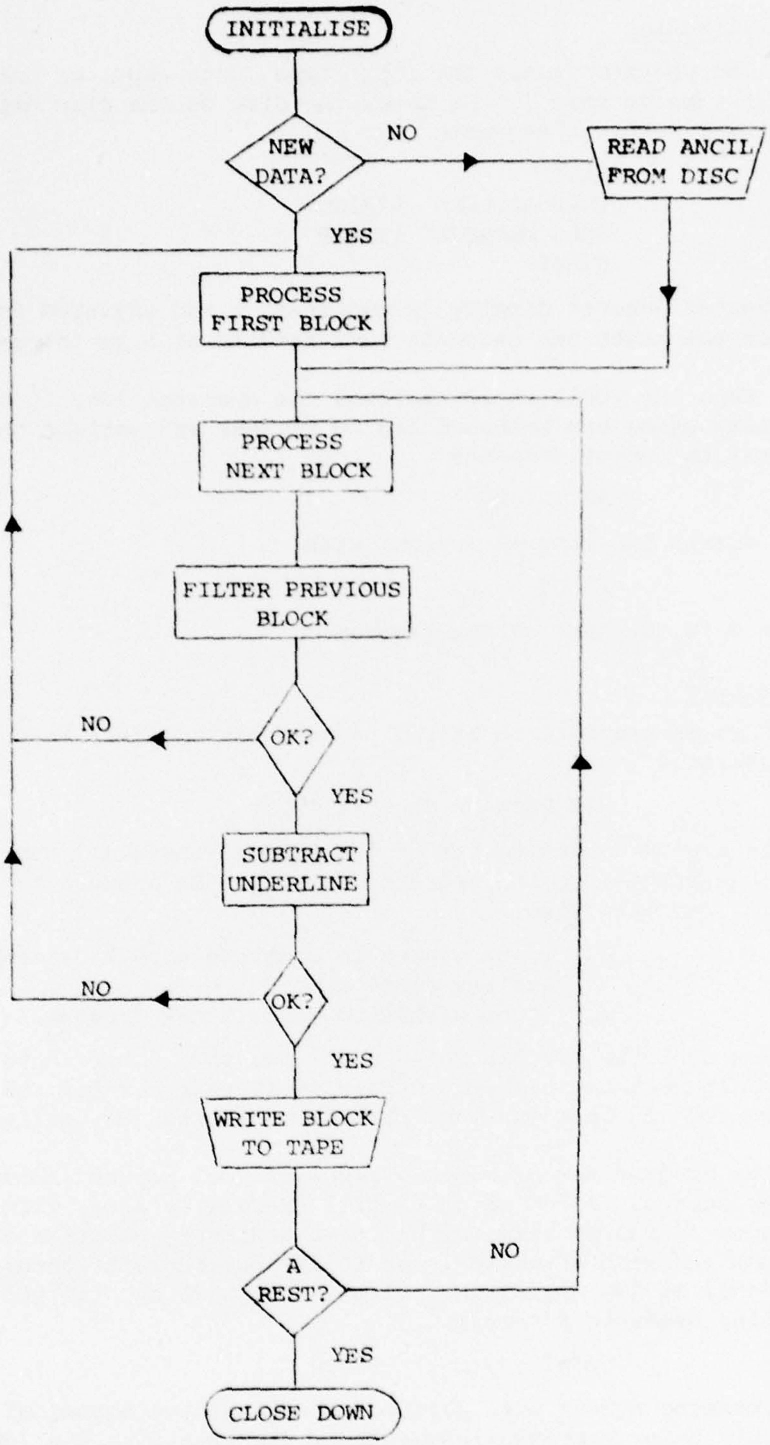

Figure 2. Simplified block diagram of the main program.

Initialisation

The operator loads the input tape (data tape) on unit 0 and the output tape on unit 1. He loads the disc on the disc drive and initialises the PDP11 with the sequence

```
HALT
LOAD ADDRESS  173100
LOAD ADDRESS  177406
START
```

The Vector General display is switched on and adjusted for high gain (since the light pen responds more readily at high intensity).

When the PDP11 is initialised the operator logs in on the Teletype keyboard using his password (eg LO 61,166) and assigns the Vector General to output stream 8

```
AS VG:,8↵
```

then starts the program running with

```
RU G↵
```

where G is the name of the program.

Operations

1. Upon initialisation of the program the operator is presented with the question

```
NEW DATA OR OLD? (N OR O)
```

in the top left hand corner of the screen (where all comments or questions from the program to the operator appear). He answers on the Vector General keyboard with

O↵ if he wishes to continue with a dataset begun in an earlier session, or
N↵ if he wishes to start a new dataset.

For new data the program sets up its own ramp function to fill up one end of the Hamming filter. Otherwise it searches for the last block to be treated and uses the last part of that block to fill up the filter.

2. The program now presents a block of 2000 points (about 2 km of data) on the screen, scaled so as to fill the active area, with a y=0 line included (although this may be in an arbitrary position due to wrap-around) and with a vertical bar in the bottom left corner representing the scale of 200 digitising units (about 0.86 m). If the profile has any line segments for which

$$y(n) - y(n-1) \geq 200$$

the tracking square will position itself at the bottom of the screen directly below the jump in question, starting with the jump nearest the start of the block. The operator now has the option of accepting the program's judgement, ie of having the jump removed on the grounds that it is a phase jump or a noise spike. To do this he just types

↵

If he thinks, however, that the jump represents real data (part of a ridge) or that it can be better removed by the "G" facility (see 3 below) he can reject the program's decision by typing

N ↲

The tracking square now moves on to the next jump until all jumps are exhausted. An instantaneous phase jump has an amplitude of 10 feet (712 digitising units); we use 200 as the criterion because jumps seldom occur instantaneously.

3. If there are now critical jumps in the block, or when the operator has dealt with all those that exist, the program presents on the screen

TRY G R X OR O   nnn

where nnn is the number of the block (for information purposes). If the operator types

R ↲

he can see the results of his operations in stage (2) since this <u>redraws</u> the block in its updated state, ie with the results of all manipulations to date. R can be used at any point during this stage of processing to examine the latest state of the data.

The "G" facility deals with any undesirable features of the data that still remain. It replaces a short length of data with a horizontal straight line segment. It can therefore be used to draw a line to replace a noise spike, or else to take out a phase jump which occupies a finite number of data points instead of being instantaneous. To use this facility the operator uses the light pen to move the tracking square to the left side of the proposed line segment and types

G ↲

He then moves to the right side of the proposed segment and types

↲

For typical applications, see EXAMPLES.

If the operator is not satisfied with the alterations that he has made he can type

X ↲

which redraws the block in its original unmodified state, rejecting all of the operator's manipulations to date.

When the operator is finally satisfied that the block now consists only of real data, with spikes and jumps removed, and subject only to aircraft motion (which should appear as a smoothly trending curve) he types

O ↲

The program now proceeds to the next stage.

4. It redraws the block once more and presents the message

UNDERLINE

It also presents the tracking square at the extreme left side of the screen on the y-value that it had reached while underlining the preceding block. If the operator is dissatisfied with this as a starting position for his underlining, he moves the square to a better position at the start of the block and types

B↲

Otherwise he simply moves the square along (using the light pen), stopping at points which he feels are representative of the "level ice surface" and typing

↲

In this way he constructs a piecewise straight line profile of the surface of level ice. To recognise this surface requires judgement, experience and sometimes imagination; a prior study of some vertical aerial photographs of the Arctic ice cover is an advantage. The main problem is not to be misled by very smooth ice (with a "roughness" which is only the noise level of the original tape recorder). This is often thin polynya ice, and the operator should try to keep a short distance above such ice, following the level of smooth floe ice (which usually has greater roughness). Narrow cracks can also be mistaken for level ice segments. The aim is to achieve consistency; ie to follow either the level of polynya ice throughout (for open icefields) or to follow the level of smooth thick ice throughout, which is more appropriate for the central Arctic where polynyas may be far apart.

If the operator is dissatisfied with his latest line segment he can type

D↲

which deletes the last segment drawn. Repeated use will delete part or whole of his line. If by mistake he does not proceed monotonically in the positive x-direction he will receive the message

MOVE ALONG

and his last segment will be disregarded. If he exceeds the maximum allowable number of segments in a line (25) the program sends

DELETE SOME

and the operator must delete some of his points and redraw the rest of the line with points further apart.

After taking his line to the extreme right of the block (with the help of a chart recorder printout, if available, to determine the continuing trend), the operator types

O↲

which sets the filter routine in action.

5. The program now filters the preceding block's underline, with the aid of the first few points in the current block to fill up the Hamming filter. When the filtering is complete (about 15 seconds) the program presents the preceding block together with its piecewise line filtered into a smooth curve, and asks

OK? Y OR N

The operator examines the smoothed underlining to see whether it appears to be a true representation of the level ice surface. If he thinks that it is, he types

Y ↲

otherwise N ↲

A rejection takes the program back to stage (4) for the preceding block; it presents this block again for underlining. The formerly current block is lost and must be processed again after the preceding block. On acceptance, the program subtracts the smoothed piecewise profile from the laser data to obtain a final laser profile, allegedly a true representation of the ice surface. It presents this again to the operator with the message

OK? (Y OR N)

Again the operator has the option of rejecting it, with the same consequences as before. If he accepts the final version, it is written onto the output tape and the program asks

A REST? (Y OR N)

If the operator wishes to wind up the session he types

Y ↲

and the program rewinds both tapes, having stored the current operating data. The operator now types

FI ↲

on the Teletype keyboard, which stops the program so that it is safe for him to remove tapes and disc.

If he wishes to process another block, he types

N ↲ on the Vector General keyboard

whereupon the next block of raw data is read in from the input tape and presented to him.

4. ILLUSTRATED EXAMPLES OF THE USE OF THE SYSTEM

In this section we show several photographs of the Vector General display screen which demonstrate aspects of the operation. Figure 3 shows the calibration signal used to eliminate errors from the tape recorder and digitisation. The profilometer output could be held at any one of three DC values corresponding to +180° offset, zero, and -180°. At least 5 seconds of each of the three signals were recorded at the beginning of each tape and whenever adjustments were made to the sensors. The left hand side of the trace shown in fig 3 has a 0° to -180° calibration signal. The average value of each offset was calculated from a print-out of the screen display, and used to establish the relationship between the unit of digitisation in the computer and the actual elevation perceived by the profilometer (the absolute accuracy of the profilometer is dependent on the stability of a quartz oscillator kept in a temperature controlled oven, and is of the order of 1 part in 10^5). Figure 3 also gives an impression of the noise level introduced into the data by the recorder and digitiser; this is just the thickness of the "line" representing the DC signal.

Figure 4 shows a typical section of heavily ridged ice from the 1976 co-operative profiling experiment. The vertical bar at the lower left corner is included by the program to give the operator an idea of the vertical scale, since the computer always re-formats the profile to occupy the whole height of the screen. The bar is 200 units (0.86 m) high. The tracking square has placed itself automatically beneath a spot where there is a jump of more than 200 units. In this case it has clearly identified the position of a 180° phase shift.

In figure 5 the operator has accepted the program's judgement and by typing ↵ has removed the jump. The jump was a simple one, and its removal reveals the presence of a high pressure ridge.

There is a second phase jump on the far right of the section. This has a finite rise time and was not found by the program. The operator moves the tracking square to the beginning and end of the jump, and by using the "G" command reduces the jump to a short horizontal line segment. Figure 6 shows the result of this operation.

Since the raw data have no further phase jumps or spikes the operator now underlines it with short line segments joining apparent minima or stretches of smooth ice (figure 7). Figure 8 shows the result of filtering this piecewise line into a smooth curve. The operator accepts the result and subtracts the smoothed line from the profile. Figure 9 shows the final result.

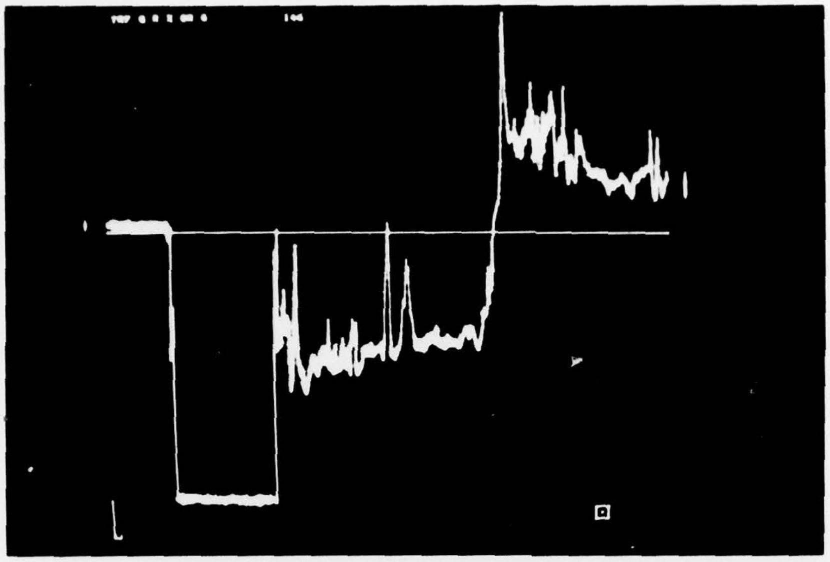

Figure 3

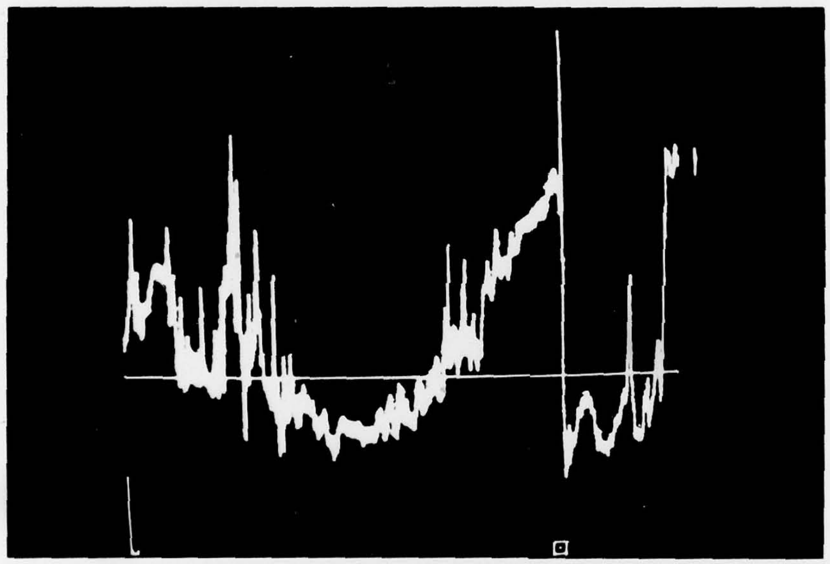

Figure 4

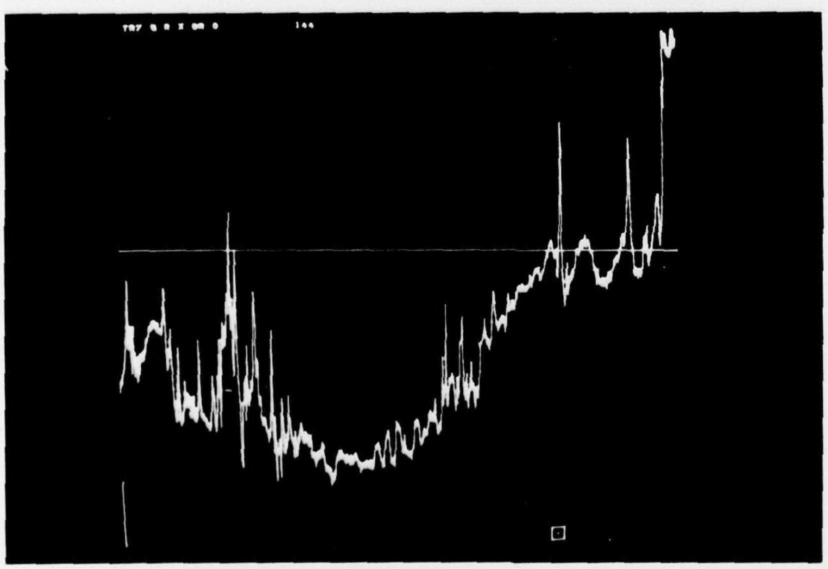

Figure 5

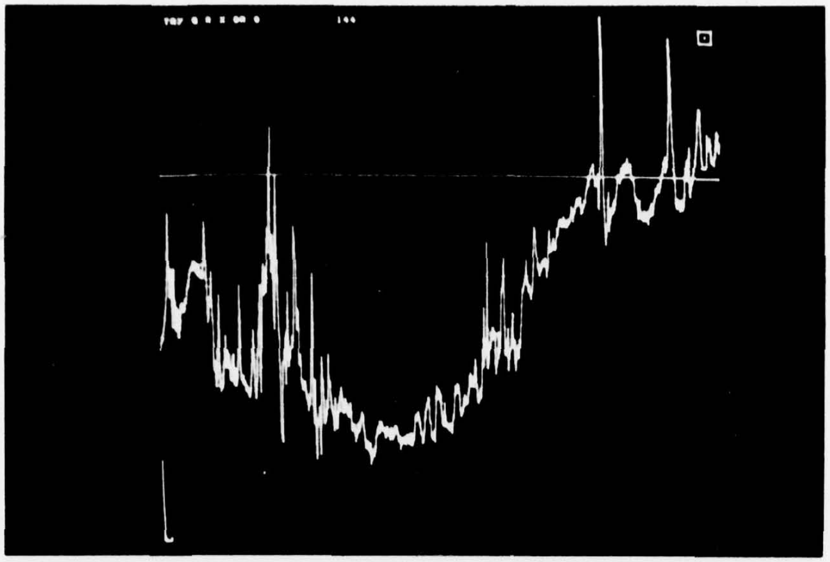

Figure 6

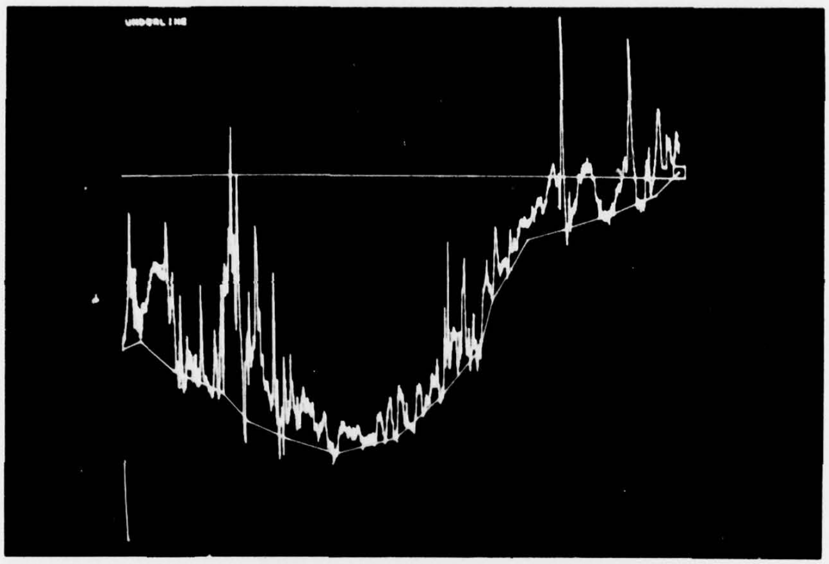

Figure 7

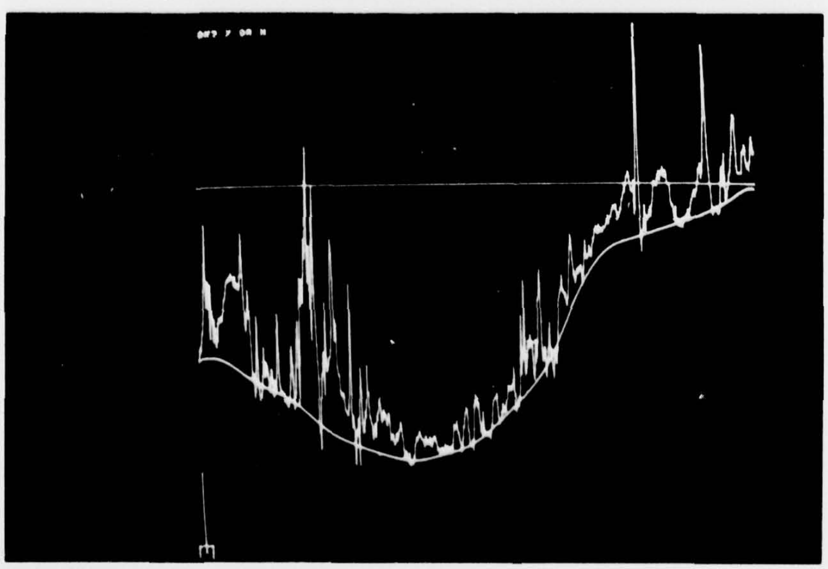

Figure 8

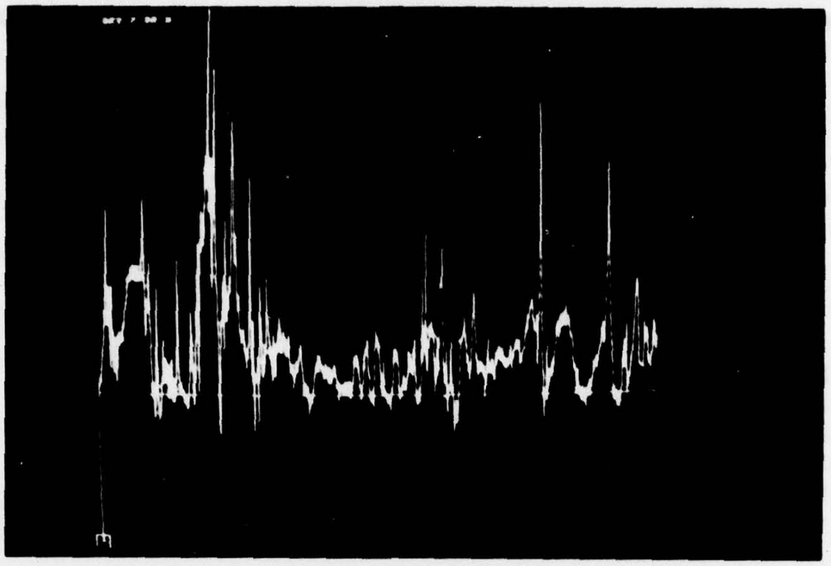

Figure 9

5. LISTING OF THE PROGRAM

The program listing is followed by block diagrams of the two most important subroutines (figures 10 and 11); a block diagram of the main program was given as fig 2.

```
C*********************************************************************
C*********************************************************************
C   MAIN PROGRAM FOR THE CORRECTION OF LASER PROFILES OF SEA ICE
C*********************************************************************
      COMMON /A/W(4)/B/TC(2)
      INTEGER ICE(2000),PTS(2000),PICBUF(4200),BSTR(101),ESTR(101)
      INTEGER ANCIL(105),V,OFSET
      EQUIVALENCE (N,ANCIL(1)),(OFSET,ANCIL(2)),(LPT,ANCIL(3))
      EQUIVALENCE(M,ANCIL(4)),(BSTR(1),ANCIL(5)),(BSTR(101),PTS(1))
      EQUIVALENCE(PTS(2000),ESTR(1))
C*********************************************************************
C   INITIALIZATION
C*********************************************************************
      DEFINE FILE 1(1,105,U,NAN)
      DEFINE FILE 2(2,2000,U,NPTS)
      DEFINE FILE 3(2,2000,U,NICE)
      CALL NITDEV(8)
      CALL NITBUF(PICBUF,4200)
      CALL MTRWND(0)
      CALL MTRWND(1)
      CALL HAMTAB
C*********************************************************************
C   STARTING AFRESH OR CONTINUING ?
C*********************************************************************
      M=0
1     CALL BGNPIC(3)
      WRITE(8,100)
100   FORMAT(1H1,'NEW DATA OR OLD? (N OR O)')
      CALL ENDPIC
      READ(8,101) V
101   FORMAT(A1)
      CALL DELPIC(3)
      IF(V.EQ.'N') GOTO 2
C*********************************************************************
C   CONTINUING
C*********************************************************************
      READ(1'1) ANCIL
      CALL MTSKIP(1,M,IERR,IEOF)
      GOTO 3
C*********************************************************************
C   STARTING AFRESH
C*********************************************************************
2     CALL BGNPIC(3)
      WRITE(8,106)
106   FORMAT(1H1,'STARTING AT BLOCK ')
      CALL ENDPIC
      READ(8,105) N
105   FORMAT(I4)
      CALL DELPIC(3)
      OFSET=0
      LPT=0
21    CALL TAPIN(ICE,N,IEOF)
      CALL CORREK(ICE,OFSET,N)
      CALL UNDERL(PTS,LPT)
      CALL HAMBGN(BSTR)
      K=MOD(N,2)+1
      WRITE(1'1) ANCIL
      WRITE(2'K) PTS
      WRITE(3'K) ICE
C*********************************************************************
C   PROCESS NEXT BLOCK
C*********************************************************************
```

THIS PAGE IS BEST QUALITY PRACTICABLE
FROM COPY FURNISHED TO DDC

```
3       N=N+1
        CALL TAPIN(ICE,N,IEOF)
        IF(IEOF.EQ.0) GOTO 4
        CALL HAMEND(ESTR)
        GOTO 5
4       DO 41 K=1,2000
41      ICE(K)=ICE(K)+OFSET
        CALL CORREK(ICE,OFSET,N)
        CALL UNDERL(PTS,LPT)
        K=MOD(N,2)+1
        WRITE(2'K) PTS
        WRITE(3'K) ICE
        CALL HAMCOP(PTS,ESTR)
5       K=N-1
        K=MOD(K,2)+1
        READ(2'K) PTS
        READ(3'K) ICE
        CALL HAMFIL(BSTR)
        CALL WINDOW(ICE,1,2000)
        CALL DISPLA(ICE,1,2000,2,1)
        CALL DISPLA(PTS,1,2000,2,2)
C*********************************************************************
C   IS THE UNDERLINE OK ?
C*********************************************************************
6       CALL BGNPIC(3)
        WRITE(8,107)
107     FORMAT(1H1,'OK? Y OR N ')
        CALL ENDPIC
        READ(8,101) V
        CALL DELPIC(1)
        CALL DELPIC(2)
        CALL DELPIC(3)
        CALL DELPIC(99)
        IF(V.EQ.'N') GOTO 8
        DO 61 K=1,2000
61      ICE(K)=ICE(K)-PTS(K)
        CALL WINDOW(ICE,1,2000)
        CALL DISPLA(ICE,1,2000,1,1)
C*********************************************************************
C   IS THE RESULT GOOD ENOUGH TO GO ON TAPE ?
C*********************************************************************
        CALL BGNPIC(3)
        WRITE(8,107)
        CALL ENDPIC
7       READ(8,101) V
        CALL DELPIC(1)
        CALL DELPIC(3)
        CALL DELPIC(99)
        IF(V.EQ.'N') GOTO 8
        CALL MTWRT(1,ICE,2000,IERR)
        M=M+1
        WRITE(1'1) ANCIL
        IF(IEOF.NE.0) GOTO 11
        CALL BGNPIC(3)
        WRITE(8,103)
103     FORMAT(1H1,'A REST? (Y OR N)')
        CALL ENDPIC
10      READ(8,101) V
        IF(V.NE.'Y'.AND.V.NE.'N') GOTO 10
        CALL DELPIC(3)
        IF(V.EQ.'N') GOTO 3
```

```
C***********************************************************************
C     CLOSE OUTPUT TAPE AND REWIND BOTH BEFORE STOPPING
C***********************************************************************
11      IF(IEOF.NE.0) WRITE(6,104)
104     FORMAT(' END OF DATA')
        CALL MTWTM(1)
        CALL MTWTM(1)
        CALL MTRWND(1)
        CALL MTRWND(0)
        STOP
C***********************************************************************
C     NOT GOOD ENOUGH. GO BACK AND DO IT AGAIN
C***********************************************************************
8       N=N-1
        GOTO 21
        END
C***********************************************************************
C***********************************************************************
C     SUBROUTINE FOR READING A BLOCK FROM THE INPUT TAPE
C***********************************************************************
        SUBROUTINE TAPIN(ICE,N,IEOF)
        INTEGER ICE(1)
        DATA NB/1/
        IF(N.EQ.NB) GOTO 1
        K=N-NB
        CALL MTSKIP(0,K,IERR,IEOF)
        IF(IERR.EQ.0) GOTO 11
        WRITE(6,100) N
100     FORMAT(' ERROR ON SKIPPING TO BLOCK',I5)
        STOP
11      IF(IEOF.NE.0) RETURN
1       NB=N+1
        CALL MTREAD(0,ICE,2000,IERR,IEOF)
        DO 2 I=1,2000
2       ICE(I)=-ICE(I)
        IF(IERR.EQ.0) RETURN
        WRITE(6,101) N
101     FORMAT(' ERROR ON READING BLOCK',I5)
        STOP
        END
C***********************************************************************
C***********************************************************************
C     SUBROUTINE TO CORRECT DATA FOR PHASE SHIFTS AND NOISE SPIKES
C***********************************************************************
        SUBROUTINE CORREK(ICE,OFSET,N)
        COMMON/A/W(4)/B/TC(2)
        DIMENSION X(2),Y(2)
        INTEGER IED(2),ICE(1),OFSET,L,H
        DATA IED/2,-1/
        KA=OFSET
        CALL WINDOW(ICE,1,2000)
        CALL DISPLA(ICE,1,2000,1,1)
45      DO 44 I=1,1999
        J=ICE(I+1)-ICE(I)
        IF(J.LT.200.AND.J.GT.-200) GOTO 44
        TC(1)=(I-W(1))/(W(2)-W(1))
        TC(2)=W(3)
        CALL WRITOL(201,0,0,TC,2)
        K=INWAIT(-1.,IED,IDA,1)
        IF(IDA.NE.' ') GOTO 44
        JJ=I+1
```

```
      DO 23 K=JJ,2000
23    ICE(K)=ICE(K)-J
      OFSET=OFSET-J
44    CONTINUE
      CALL BGNPIC(3)
      WRITE(8,101) N
101   FORMAT(1H1,'TRY G R X OR O',10X,I4)
      CALL ENDPIC
1     I=INWAIT(-1.,IED,IDA,1)
      IF(IDA.NE.'G') GOTO 2
      CALL REATOL(201,0,0,TC,2)
      X(1)=TC(1)*(W(2)-W(1))+W(1)
      Y(1)=TC(2)*(W(4)-W(3))+W(3)
      K1=IFIX(X(1))
      I=INWAIT(-1.,IED)
      CALL REATOL(201,0,0,TC,2)
      X(2)=TC(1)*(W(2)-W(1))+W(1)
      Y(2)=TC(2)*(W(4)-W(3))+W(3)
      K2=IFIX(X(2))
      IV=IFIX(Y(1)-Y(2))
      DO 11 KK=K1,K2
11    ICE(KK)=ICE(K1)
      K2=K2+1
      DO 12 KK=K2,2000
12    ICE(KK)=ICE(KK)+IV
      OFSET=OFSET+IV
      GOTO 1
2     IF(IDA.NE.'R'.AND.IDA.NE.'X'.AND.IDA.NE.'O') GOTO 1
      IF(IDA.NE.'X') GOTO 22
      CALL DELPIC(3)
      CALL TAPIN(ICE,N,IEOF)
      OFSET=KA
      DO 21 KK=1,2000
21    ICE(KK)=ICE(KK)+OFSET
22    CALL DELPIC(99)
      CALL WINDOW(ICE,1,2000)
      CALL DELPIC(1)
      CALL DISPLA(ICE,1,2000,1,1)
      IF(IDA.EQ.'X')GOTO 45
      IF(IDA.EQ.'O')RETURN
      GOTO 1
      END
C*********************************************************************
C*********************************************************************
C   SUBROUTINE TO UNDERLINE THE DATA
C*********************************************************************
      SUBROUTINE UNDERL(PTS,LPT)
      COMMON /A/W(4)/B/TC(2)
      INTEGER IED(2),PTS(1),ITMP(100)
      DATA IED/2,-1/MAX/48/
      CALL DELPIC(3)
      CALL BGNPIC(3)
      WRITE(8,100)
100   FORMAT(1H1,'UNDERLINE')
      CALL ENDPIC
      CALL BGNPIC(2)
      CALL ENDPIC
6     ITMP(1)=IFIX(W(1))
      ITMP(2)=LPT
      K=3
      CALL DRAW(ITMP,K)
1     I=INWAIT(-1.,IED,IDA,1)
```

```
      IF(IDA.EQ.'O') GOTO 4
      IF(IDA.NE.' ') GOTO 2
      IF(K.LT.MAX) GOTO 11
      CALL DELPIC(3)
      CALL BGNPIC(3)
      WRITE(8,101)
101   FORMAT(1H1,'DELETE SOME')
      CALL ENDPIC
      GOTO 13
11    CALL REATOL(201,0,0,TC,2)
      ITMP(K)=IFIX(TC(1)*(W(2)-W(1))+W(1))
      ITMP(K+1)=IFIX(TC(2)*(W(4)-W(3))+W(3))
      IF(ITMP(K).GT.ITMP(K-2)+1) GOTO 12
      CALL DELPIC(3)
      CALL BGNPIC(3)
      WRITE(8,102)
102   FORMAT(1H1,'MOVE ALONG')
      CALL ENDPIC
      GOTO 1
12    K=K+2
      CALL DRAW(ITMP,K)
13    GOTO 1
2     IF(IDA.NE.'D') GOTO 3
      IF(K.EQ.3) GOTO 1
      K=K-2
      CALL DRAW(ITMP,K)
21    GOTO 1
3     IF(IDA.NE.'B') GOTO 35
      CALL REATOL(201,0,0,TC,2)
      LPT=IFIX(TC(2)*(W(4)-W(3))+W(3))
      GOTO 6
35    CALL DELPIC(3)
      CALL BGNPIC(3)
      WRITE(8,103)
103   FORMAT(1H1,'TRY B,I,D, OR O')
      CALL ENDPIC
      GOTO 1
4     CALL DELPIC(1)
      CALL DELPIC(2)
      CALL DELPIC(3)
      CALL DELPIC(99)
      TC(1)=0.
      TC(2)=0.
      CALL WRITOL(201,0,0,TC,2)
      ITMP(K)=2000
      ITMP(K+1)=ITMP(K-1)
      K=K-2
      DO 41 I=1,K,2
      IX=ITMP(I+2)-ITMP(I)
      IY=ITMP(I+3)-ITMP(I+1)
      L1=ITMP(I)
      L2=ITMP(I+2)
      DO 41 L=L1,L2
      IA=L-ITMP(I)
41    PTS(L)=ITMP(I+1)+IFIX(FLOAT(IY)*FLOAT(IA)/FLOAT(IX))
      LPT=PTS(2000)
      RETURN
      END
```

****

```
C************************************************************************
C     SUBROUTINE TO DRAW PART OF AN UNDERLINE
C************************************************************************
      SUBROUTINE DRAW(ITMP,K)
      COMMON /A/W(4)/B/TC(2)
      INTEGER ITMP(1)
      CALL DELPIC(2)
      CALL BGNPIC(2)
      CALL LINI(ITMP(1),ITMP(2),0)
      CALL LINIR(0,0,4)
      N=K-2
      IF(K.EQ.3) GOTO 2
      DO 1 I=3,N,2
1     CALL LINI(ITMP(I),ITMP(I+1))
2     CALL ENDPIC
      TC(1)=(FLOAT(ITMP(N))-W(1))/(W(2)-W(1))
      TC(2)=(FLOAT(ITMP(N+1))-W(3))/(W(4)-W(3))
      CALL WRITOL(201,0,0,TC,2)
      RETURN
      END
C************************************************************************
C************************************************************************
C     SUBROUTINE TO DISPLAY DATA
C************************************************************************
      SUBROUTINE DISPLA(IA,L,H,KBY,NPIC)
      DIMENSION IA(1)
      INTEGER L,H
      CALL BGNPIC(NPIC)
      CALL LINI(L,IA(L),0)
      DO 1 I=L,H,KBY
1     CALL LINI(I,IA(I))
      CALL ENDPIC
      RETURN
      END
C************************************************************************
C************************************************************************
C     SUBROUTINE TO WINDOW THE DATA
C************************************************************************
      SUBROUTINE WINDOW(IA,L,H)
      COMMON /A/W(4)
      INTEGER IA(1),YL,YH,L,H
      DATA IVAL/200/
      YL=IA(L)
      YH=IA(L)
      DO 1 K=L,H
      IF(IA(K).LT.YL) YL=IA(K)
      IF(IA(K).GT.YH) YH=IA(K)
1     CONTINUE
      W(1)=L
      W(2)=H
      W(3)=YL-200.
      W(4)=YH
      CALL WINDW(W)
      CALL BGNPIC(99)
      CALL LINE(W(1),W(3),0)
      CALL LINIR(0,IVAL)
      CALL LINE(W(1),0,0)
      CALL LINE(W(2),0)
      CALL ENDPIC
      RETURN
      END
C************************************************************************
```

```
C*********************************************************************
C   SUBROUTINE TO TABULATE COEFFICIENTS FOR THE FILTER
C*********************************************************************
      SUBROUTINE HAMTAB
      COMMON /C/TABL(200)
      DATA P/3.1415926536/
      A=-200.
      DO 1 K=1,200
      TABL(K)=0.54+0.46*COS(P*A/200.)
1     A=A+2
      RETURN
      END
C*********************************************************************
C*********************************************************************
C   START RAMP FOR FILTER
C*********************************************************************
      SUBROUTINE HAMBGN(BSTR)
      INTEGER BSTR(1)
      DO 1 I=1,100
      AI=I
1     BSTR(I)=BSTR(101)
      RETURN
      END
C*********************************************************************
C*********************************************************************
C   END RAMP FOR FILTER
C*********************************************************************
      SUBROUTINE HAMEND(ESTR)
      INTEGER ESTR(1)
      DO 1 I=1,100
      ESTR(I+1)=(100.-FLOAT(I))*ESTR(1)/100.
1     CONTINUE
      RETURN
      END
C*********************************************************************
C*********************************************************************
C   COPY ROUTINE FOR FILTER
C*********************************************************************
      SUBROUTINE HAMCOP(PTS,ESTR)
      INTEGER PTS(1),ESTR(1)
      DO 1 I=1,100
1     ESTR(I+1)=PTS(I)
      RETURN
      END

C*********************************************************************
C*********************************************************************
C   SUBROUTINE TO FILTER THE UNDERLINE. IN PRACTICE THIS WAS
C   REPLACED BY AN EQUIVALENT ASSEMBLER ROUTINE.
C*********************************************************************
      SUBROUTINE HAMFIL(BSTR)
      COMMON /C/TABL(200)
      INTEGER BSTR(1)
      DO 1 I=1,2000
      SUM=0
      DO 2 K=1,200
2     SUM=SUM+BSTR(I+K-1)*TABL(K)
1     BSTR(I)=SUM/108.
      DO 3 I=1,2100
3     BSTR(2201-I)=BSTR(2101-I)
      RETURN
      END
.....
```

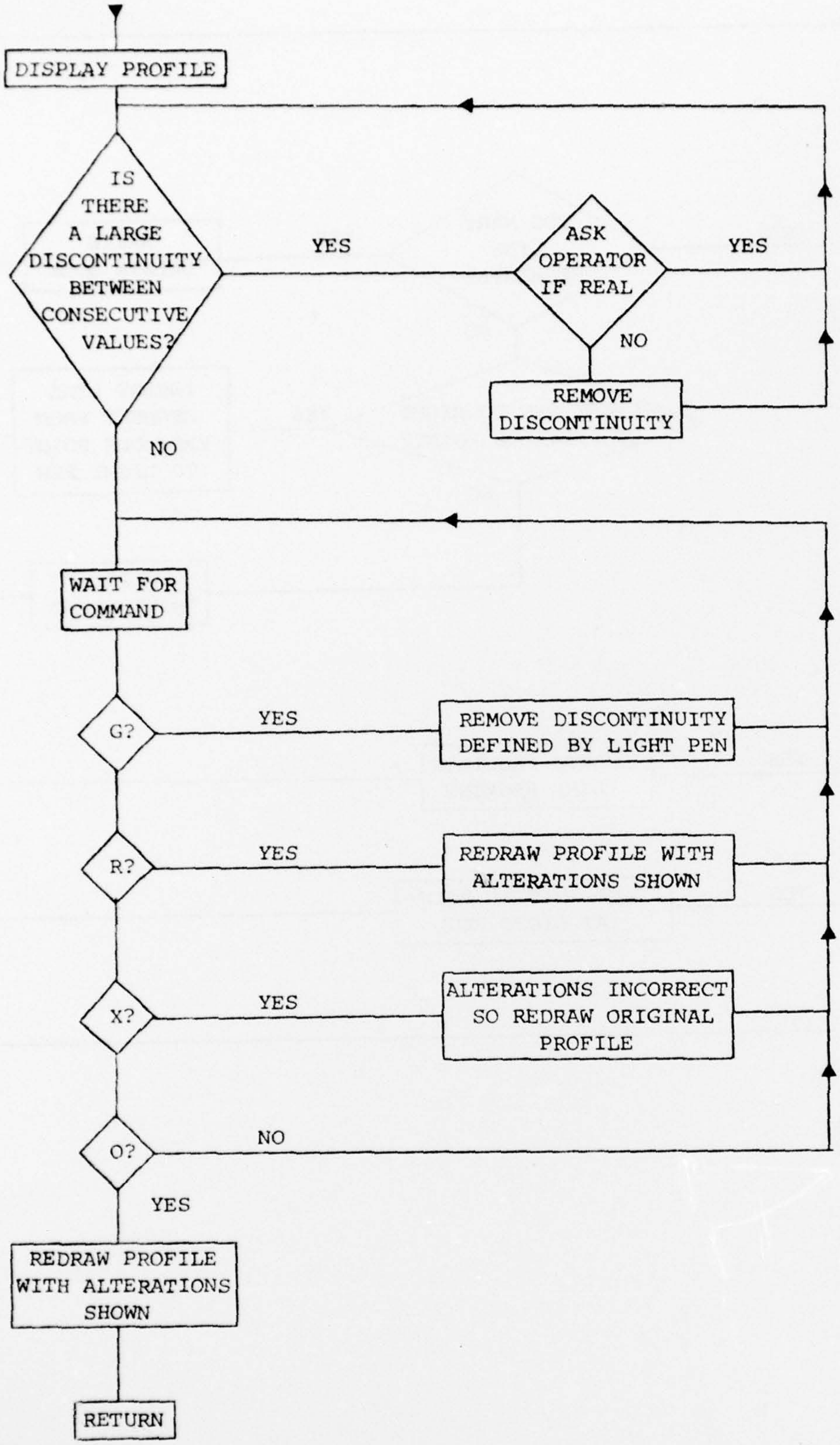

Figure 10. CORREK subroutine

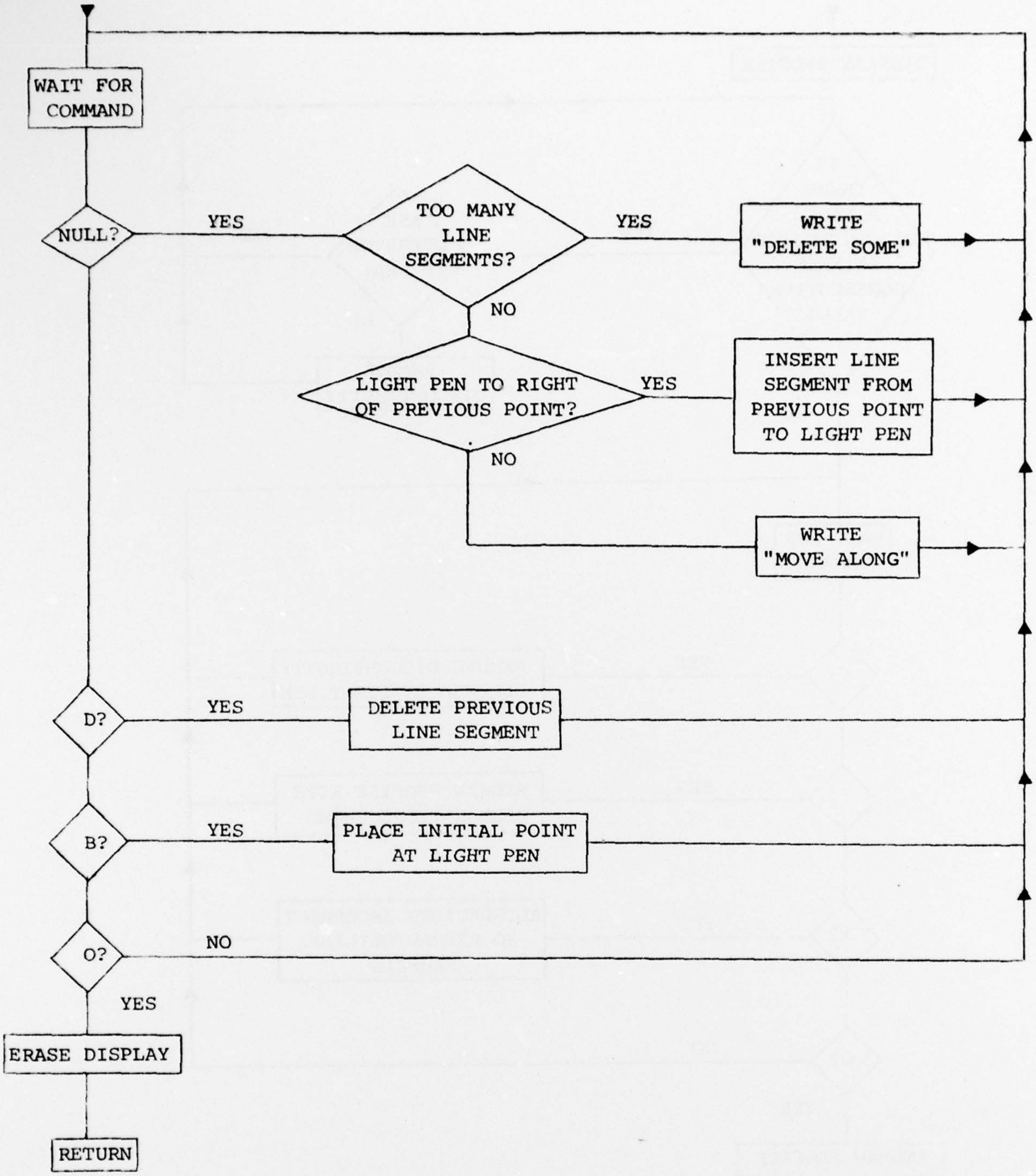

Figure 11. UNDERL subroutine.

ACKNOWLEDGEMENTS

It is a pleasure to acknowledge the support of the Office of Naval Research under contract N00014-76-C-0660. This processing system has developed from a program written at Defence Research Establishment, Ottawa, to which the contribution of C J Brochu is gratefully acknowledged.

REFERENCES

Banke, E G, S D Smith and R J Anderson (1976). Recent measurements of wind stress on Arctic sea ice. J. Fish. Res. Bd. Canada, 33 (10), 2307-2317.

Diachok, O I (1975). A simple geometrical/statistical model of sea ice ridges. Tech. Note TN 6130-5-75, Naval Oceanogr. Off., Washington DC, 20373, 15pp.

Dunbar, M and R T Lowry (1974). Remote sensing of sea ice in Nares Strait and the Arctic Ocean, March 1973. Proc. 2nd Canadian Symp. on Remote Sensing, Guelph, Ont., April 29 - May 1 1974.

Hibler, W D III (1972). Removal of aircraft altitude variation from laser profiles of the Arctic ice pack. J. Geophys. Res. 77 (30), 7190-7195.

Hibler, W D III (1975). Statistical variations in sea ice ridging and deformation rates. Proc. Ice Tech. Symp., Montreal, 9-11 April 1975. Soc. Nav. Archit. Mar. Engrs, New York, J1-J9.

Hibler, W D III, W F Weeks and S J Mock (1972). Statistical aspects of sea ice ridge distributions. J. Geophys. Res., 77 (30), 5954-5970.

Hibler, W D III, S J Mock and W B Tucker (1974). Classification and variation of sea ice ridging in the western Arctic Basin. J. Geophys. Res., 79 (18), 2735-2743.

Ketchum, R D Jr (1971). Airborne laser profiling of the Arctic pack ice. Remote Sensing of Environment, 2, 41-52.

Kozo, T L and O I Diachok (1973). Spatial variability of topside and bottomside ice roughness and its relevance to underside acoustic reflection loss. AIDJEX Bull. 19, 113-121.

Lowry, R T and C J Brochu (1976). An interactive correction and analysis system for airborne laser profiles of sea ice. Presented at Canadian Aeronautics & Space Institute (CASI) Aerospace Electronics Symp., Banff, Feb 1976 (CASI J Remote Sensing, in press)

Maykut, G A (1976). Energy exchange over young sea ice in the central Arctic. AIDJEX Bull, 31, 45-74.

Tooma, S G Jr and W B Tucker (1973). Statistical comparison of airborne laser and stereophotogrammetric sea ice profiles. Remote Sensing of Environment, 2 (4), 261-272.

Tucker, W B and V H Westhall (1973). Arctic sea ice ridge frequency distributions derived from laser profiles. AIDJEX Bull. 21, 171-180.

Wadhams, P (1975). Airborne laser profiling of swell in an open ice field. J. Geophys. Res. 80 (33), 4520-4528.

Wadhams, P (1976). Sea ice topography in the Beaufort Sea and its effect on oil containment. AIDJEX Bull. 33, 1-52.

Wadhams, P (1977). A comparison of sonar and laser profiles along corresponding tracks in the Arctic Ocean. Presented at AIDJEX/ICES Symposium on Sea Ice Processes and Models, Seattle, 6-9 Sept. 1977.

Wadhams, P and R T Lowry (1977). A joint topside/bottomside remote sensing equipment on Arctic sea ice. Proc. 4th Canadian Symposium on Remote Sensing, Québec, 16-18 May 1977. Canadian Remote Sensing Soc, 407-423.

Welsh, J P and W B Tucker (1971). Sea ice laser statistics. Proc. 7th Int. Symp. on Remote Sensing of Environment, Ann Arbor, Willow Run Labs, U. of Michigan, 2, 1165-1175.